A new mobility-based clustering algorithm for Vehicular Ad Hoc Networks (VANETs)

Mengying Ren*, Lyes Khoukhi*, Houda Labiod†, Jun Zhang† and Veronique Veque‡
*University of Technology of Troyes, France
lyes.khoukhi@utt.fr
†Telecom ParisTech, University of Paris-Saclay, France
{mengying.ren, houda.labiod, jun.zhang}@telecom-paristech.fr
‡University of Paris-Sud, France
veronique.veque@u-psud.fr

***Abstract*—Clustering in vehicular ad hoc networks (VANETs) is a challenging issue due to the highly dynamic vehicle mobility and frequent communication disconnections problems. Recent years' research have proven that mobility-based clustering mechanisms considering speed, moving direction, position, destination and density, were more effective in improving cluster stability. In this paper, we propose a new mobility-based and stability-based clustering algorithm (MSCA) for urban city scenario, which makes use of vehicle's moving direction, relative position and link lifetime estimation. We evaluate the performance of our proposed algorithm in terms of changing maximum lane speed and traffic flow rate. Our proposed algorithm performs well in terms of average cluster head lifetime and average number of clusters.**

***Index Terms*—Vehicular Ad Hoc Networks, clustering algorithm, wireless communication, one-hop.**

I. Introduction

With the rapid development of automotive manufacturing, vehicles are becoming more and more intelligent and powerful. Vehicles now have the capability to communicate with other vehicles directly in a V2V (Vehicle-to-vehicle) manner or indirectly using the existing infrastructure alongside the road in a V2I (Vehicle-to-Infrastructure) or I2V (Infrastructure-to-Vehicle) manner [1]. Applications, including safety and non-safety applications, in vehicular ad hoc networks (VANETs) are dependent on these V2V, V2I or I2V communication types. Among of these applications, the most investigated and deployed application is safety application. A classification for road safety applications is given in the ETSI standard, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions" [2]. Information dissemination in VANETs usually requires low latency of transmission and high accuracy of information. A protocol called SCRP [3] was proposed to reduce the end-to-end delay by selecting proper routing paths. However, due to large number of mobile vehicles and lack of routers in VANETs, a flat network will cause scalability problems. One efficient solution to solve this problem is dividing vehicles into different groups, called clusters, according to some predefined criteria [9].

A massive number of protocols were proposed based on clustering algorithm in Mobile Ad Hoc Networks (MANETs) ([5], [6], etc.), in order to save the network resource and to increase network efficiency. In past years, researchers began to apply clustering algorithm in VANETs. According to the predictable mobility and predefined road topology in VANETs, researchers find that mobility-based metrics are more applicable for VANETs, such as vehicle moving direction, relative position, average speed, similar destination and vehicle density ([7], [8], etc.). Vehicles having similar mobility pattern are more likely to stay in the same cluster, which can enhance the cluster stability. A detailed classification of clustering protocols in VANETs is presented in [9].

In this paper, we propose a clustering algorithm aiming at providing a more stable backbone for future information dissemination in VANETs. To improve the information dissemination efficiency on the road, the proposed clustering algorithm should present a good cluster stability and low overhead. We evaluate cluster stability from the following aspects: average cluster head lifetime and average number of clusters. Mobility metrics, considered in our proposed algorithm includes vehicle's position, moving direction and velocity information. In addition, we propose an estimated connection time parameter, called Link Lifetime Estimation (LLT), to choose the gateway node in a cluster. Our simulation studies the influence of different cluster parameters on our cluster stability. We try to generate as much of real vehicle traffic scenarios as possible by using SUMO (Simulation of Urban MObility) [10].

The rest of the paper is organized as follows: Section II discusses the related work in VANET clustering. Section III presents our proposed clustering algorithm from the aspects of cluster head selection, cluster formation and cluster maintenance. Section IV presents the simulation environment and the performance analysis of our protocol. Section V concludes the paper and briefly presents our future work.

II. Related Work

A cluster is a virtual group which contains at least one cluster head (CH) and multiple cluster members (CM). Several research have been presented in VANETs dealing with vehicle

978-1-5090-0223-8/16/$31.00 © 2016 European Union

mobility metrics and cluster formation mechanisms. Some clustering mechanisms are introduced in this section.

A simple and direct way to choose a CH is selecting the first vehicle moving in a certain direction. Cluster like platooning in CONVOY [11] under highway scenario selected the first vehicle as a CH. Vehicles within the predefined maximum length to CH will be grouped together, which construct a multi-hop cluster. MC-DRIVE [12] proposed a direction-based clustering algorithm for intersection area. The first vehicle moving in a certain direction was selected as CH and clusters are formed in one-hop based on CHs' transmission range. However, these mechanisms are only suitable for simple road topology, like straight highway and intersection.

Instead of simply choosing the first vehicle as CH, most clustering mechanisms prefer calculating the stability of a node to its surroundings. MOBIC [13] was the first article proposing aggregate mobility. It was originally proposed for MANETs, but also works for VANETs. Each node calculates its relative mobilities to all of its neighbors based on Received Signal Strength (RSS). The node with the lowest aggregate mobility is chosen as the CH. Similar to MOBIC, the New-ALM [14] also chooses a node with less variance relative to its surroundings as a CH. Instead of using the RSS parameter, New-ALM calculated relative distance between to nodes. Later, to improve the cluster stability, the paper [15] proposed a k-hop clustering. K-hop relative mobility was based on the ratio of packet delivery delay of two consecutive packets. PPC [7] is also a multi-hop clustering mechanism which is based on vehicles' speed variations and the predicted traveling time. Vehicle's relative stability value "Eligibility" decreases exponentially with the increased speed deviation. APROVE [16] is based on a data clustering technique, Affinity Propagation [17]. Each vehicle sends hello messages periodically, including availability and responsibility messages. Vehicles' relative distance, position and prediction position of near future are used in APROVE. Vehicle with highest sum of availability and responsibility value is selected as a CH. A cluster contention time (CCI) is proposed when two CHs encounter each other in order to reduce the unnecessary cluster reformation.

Another clustering mechanism is based on Weighted Clustering Algorithm (WCA). The CH selection is based on the weighted sum operation. In [18], the author proposed a lane-based clustering algorithm based on vehicles' relative speed, relative position and traffic flow. Each lane can be distributed with a certain weight according to the traffic flow. VWCA [19] calculates the weighted clustering value based on the metrics: vehicle distrust value, entropy value, number of neighbors and relative position. The vehicle with the minimum weighted sum value in the neighbor is selected as CH. To improve the network connectivity, VWCA also proposed an adaptive allocation of transmission range algorithm (AATR) based on DSRC standard and vehicle density. Another weighted clustering mechanism AMACAD [20] was proposed based on vehicle's final destination, obtained by navigation system. In AMACAD, vehicles with similar destinations have higher possibility to stay in the same cluster. The weighted sum is calculated based on vehicles' relative destinations, final destinations, relative speed and current position.

Some of the above mentioned clustering algorithms mainly used the mobility and direction as cluster head selection and cluster formation metrics. All of the mentioned mechanisms are beacon-based and aim at increasing cluster stability. The performance evaluation metrics considered in these mechanisms includes: CH lifetime, CM lifetime, average status change per node, average cluster number and clustering overhead. In this paper, we aim at constructing clusters as structures for network backbone for future information dissemination especially in urban city scenarios.

III. Proposed approach

A. Scenario and assumptions

Our work focuses on proposing a new clustering algorithm based on V2V communication in urban city scenario. We leave the discussion about intersection turning prediction and information dissemination mechanisms out of the scope of this paper.

In our proposed approach, all vehicles are assumed to be equipped with a GPS which can obtain vehicle's information about position, velocity and moving direction. In addition, each vehicle can calculate relative speed with respect to its neighbors, as well as detect the distance to its neighbors.

B. Cluster definition

The objective of our proposed clustering mechanism is to improve the cluster stability and to reduce the cluster reformation times. In order to reduce the message exchanging in the cluster and to avoid the CH re-selection procedures, we choose the vehicle nearest to the central position of a cluster as a *CH*. CH's one-hop neighbors are *CMs*. Since we are considering a straight road, a cluster is represented by a rectangle on the road, shown in Fig. 1. Table I lists the notations used throughout this study.

TABLE I
Notations

Notation	Description
TR	Transmission Range
D_t	Safe Distance threshold
ΔD	Relative distance
L	Length of cluster
T_t	Timer for UN transfer to CHt
$idcluster$	Identifier of cluster
CM_table	A table maintained by CH recording its CMs
L_{mer}	Length of merged cluster
T_{merge}	Timer for cluster merging
T_{beacon}	Timer for CH to wait for beacons from its CMs
$Beacon(CM_i)$	Table for CH to record received beacons from its CMs

Due to the rapid changes of vehicle mobility, vehicles on the edge of CH's transmission range (TR) have intermittent connection with their CH, which will cause frequent cluster re-formation. To solve this problem, we introduce a "safety distance threshold", smaller than TR of CH, $D_t \leq TR$. Vehicles within D_t area have more stable connection with its CH. Therefore, the length of cluster in our proposed algorithm is defined as $L \leq 2D_t$. Fig. 1 presents two clusters, $Cluster_i$ and $Cluster_{i+1}$. Each cluster is composed of an unique CH and at least one CM. Vehicles in green represent special CMs, which can communicate with at least two CHs. These nodes are defined as gateway nodes (GW).

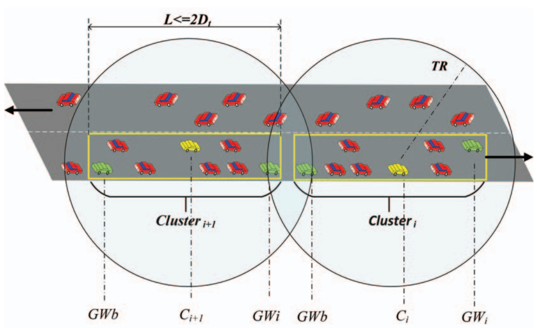


Fig. 1. Clustering mechanism (TR: Transmission Range; L: cluster length; D_t: Distance threshold; GWi, GWb: Gateway node.)

In our proposed clustering algorithm, vehicles can be one of the following 4 status: Undecided Node (UN), Cluster Head (CH), Cluster Member (CM), and Temporary Cluster Head (CHt). Status are specified in the following:

- UN: Initial state of all the vehicles, which means that the vehicle does not belong to any clusters.
- CH: The leader of the cluster, which can communicate with all its members. Each cluster has only one CH. Each cluster contains a *CM_table* which can record its CMs' information.
- CM: The vehicle which is a one-hop neighbor of a CH. A special type of CM is GW. It is responsible for inter-cluster communication and is located on the edge of the cluster. Each cluster may have two GWs: GWi moves in front of CH and GWb moves behind of CH.
- CHt: Vehicle which plays the role of CH temporarily. It only appears at the beginning of cluster formation procedure and disappears when CH is selected.

The transitions of these states are triggered by different events, presented as a state machine in Fig.2, which we will explain in the following part.

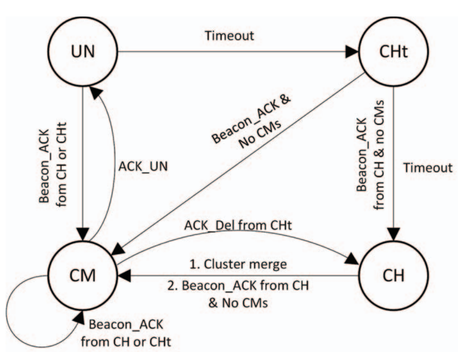


Fig. 2. Vehicle status transition

C. Cluster formation and CH selection

As we have mentioned above, the CH is at the central position of the cluster. We suppose that vehicles enter the road one by one with a certain traffic flow rate and CH is selected during the cluster formation procedure. Each vehicle sends a beacon message for each Beacon Interval. The detailed cluster formation and CH selection operations performed in our algorithm are introduced in the following steps.

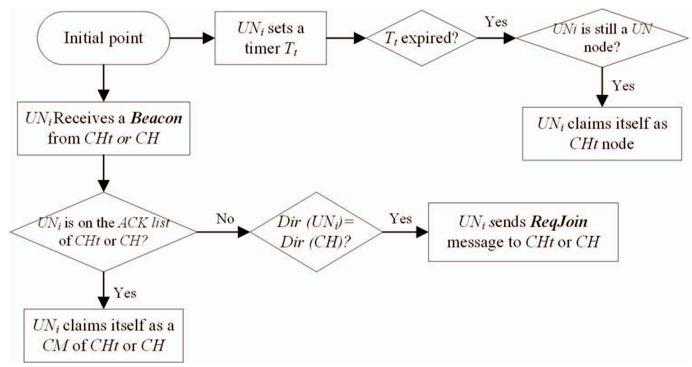


Fig. 3. Cluster formation procedure of UN

Step 1 Once a vehicle enters the road, it begins with a UN state and broadcasts a Beacon message at each Beacon Interval, containing its identifier, velocity, moving direction and position. Every vehicle, entering the road with the status UN, will set a timer T_t. When T_t is expired, if the status of this vehicle is still "UN", it claims itself as CHt and begins a cluster formation procedure.

Step 2 When vehicle UN on the road hears a *Beacon* message from a CH or CHt, it checks whether it is confirmed to join a cluster. If yes, it changes its role to CM and sets the cluster ID *idcluster*; if not, it checks whether it is moving in the same direction with the sender and then decides if it should send a *ReqJoin* message to the sender. Fig. 3 presents the cluster formation procedure of UN.

Step 3 Once a CH_i receives a *ReqJoin* message from vehicle V_j, it calculates its distance ΔD with V_j. If $\Delta D \leq D_t$ and the source vehicle is not yet its CM, CH_i adds V_j as its CM and records V_j's position to its *CM_table*. CH_i will send *ACK* notification to V_j within its next *Beacon* message, which is called *Beacon_ACK* message.

Step 4 When a CHt_i receives a *ReqJoin* message from vehicle V_j, it calculates the relative distance. If $\Delta D \leq D_t$, V_j is moving behind and is not the CM of CHt_i, CHt_i adds V_j into its *CM_table* and sends *ACK* notification within its next *Beacon* message, also called *Beacon_ACK* message; in contrast, if $\Delta D > D_t$ and CHt_i is a single node, CHt_i will change its status to CH; otherwise, if CHt_i is not a single node, CHt_i will send all its CMs to its farthest cluster member CM_k moving behind it through a

Del_CM_list message and will ask CM_k to be the CH of this cluster. CHt_i then claims itself as a CM of this new CH.

As long as the CH is selected and the cluster is well formed, CH selects the two farthest vehicles from its *CM_table* to be its GWs. However, it happens sometimes that two GW candidates have the same relative distance from their CH. To solve this problem, we introduce an estimated connection time between a CH and a CM, LLT. In our clustering scenario, the LLT can be calculated in the following equation. Obviously, a longer LLT means a more stable connection. Therefore, CH chooses the candidate vehicle which has longer LTT as its GW.

$$LLT = \frac{TR \pm |x_2 - x_1|}{|v_2 - v_1|} \quad (1)$$

Note that the TR is the transmission range of the vehicle, x_1 and x_2 are the positions of CM and CH, and v_1 and v_2 are the velocities of CM and CH, respectively.

D. Cluster maintenance

Due to the high dynamic nature of VANETs, vehicles keep joining and leaving clusters frequently, thus, causing extra maintenance overhead. In our proposed approach, cluster maintenance procedure can be described as follows.

1) Cluster merging: Our proposed algorithm allows cluster overlapping. However, when two neighbor CHs become too close and two neighboring clusters have a big overlapping area, shown in Fig. 4, cluster merging procedure is triggered. Instead of having two CHs, a single CH is selected to control all of these CMs. When the distance of two CHs is smaller than the threshold D_t, cluster merging procedure begins. To avoid frequent cluster reformation, cluster merging procedure is deferred. Instead of starting the cluster merging procedure immediately, the merging procedure is started if two CHs can always hear each other and are always within the distance D_t during T_{merge} time period. After T_{merge}, the CH_{i+1} moving behind will send a $ReqMerge$ message to the CH_i moving ahead.

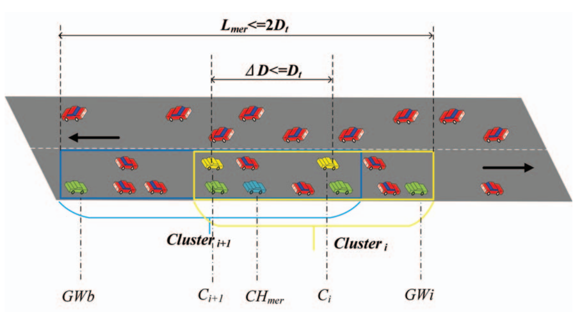


Fig. 4. Cluster merging procedure (TR: Transmission Range; L_{mer}: merged cluster length; D_t: Distance threshold.)

As long as CH_i receives $ReqMerge$ message, it firstly checks the potential merged cluster size L_{mer}. If $L_{mer} \leq 2D_t$, cluster merging is confirmed and a new CH_{mer} is selected, which is the nearest node to the central position of the merged cluster. After selecting CH_{mer}, previous CHs will send ACK_{merge} message with their CMs list to CH_{mer} and claims themselves as CMs of CH_{mer}. The new CH_{mer} adds the received CMs to its CM_table and then broadcasts a $Beacon\_ACK$ message to inform all of its CMs to change the cluster ID.

2) Leaving a cluster: In each cluster, the CH has a function of monitoring its CM_table. On the road, vehicles may enter and leave a cluster from time to time. We suppose that each CH maintains a Beacon Table to record the presence of its CMs, so that the CH can detect disconnections of its CM in time. Once the CH receives a $Beacon$ from CM_i, it checks the relative distance ΔD between CH and CM_i. If $\Delta D \leq D_t$, CH sets *Beacon(*CM_i*)* to 1; otherwise, CM_i is out of the cluster and CH discards this message. Each CH has a timer T_{beacon}: for each T_{beacon} interval, CH checks its *Beacon Table*. If *Beacon (*CM_j*)=0*, CH sends a ACK_UN message to inform CM_j that CM_j is no longer a member of CH and deletes CM_j from *CM_table* of CH.

3) CM_table and GW updating: Updating *CM_table* event is triggered when CH receives a *Beacon* message from its CM. The CM position will be updated in the *CM_table*. Therefore, every CH manages a dynamic *CM_table*. Once the *CM_table* is updated, the GWi and GWb selection functions are triggered immediately and gateway information will also be updated according to Step 6 in cluster formation procedure.

E. Important messages

Table II presents a set of important messages transmitted in our clustering algorithm. Each message is one-hop message and must contain the following parameters: message type, source ID, source state, cluster ID, X coordination, y coordination, speed and direction. Compared to a simple *Beacon* message, *Beacon_ACK* adds the CM list, and is only broadcast by CH and CHt.

TABLE II
LIST OF IMPORTANT MESSAGES

Name of the message	Source	Dissemination type
$Beacon$	UN or CM	broadcast
$Beacon\_ACK$	CH or CHt	broadcast
$ReqJoin$	Any single node	Towards a CH or CHt
$ReqMerge$	CH	Towards a CH
Del_CM_list	CH or CHt	Towards a new CH
ACK_UN	CH	Towards a CM

IV. SIMULATION

A. Simulation environment

Our proposed clustering algorithm was implemented on NS2 using the 802.11p MAC technology. The simulations were performed on a straight road with 100 vehicles. Every simulation ran for 800s. The simulations were run on different traffic traces and the performance results were averaged.

In our simulation, realistic NS2 traffic traces were generated by the open source micro-traffic simulator, SUMO [10], version 0.19.0. We set the average vehicle length of 5m, the default value in SUMO. We study a straight road with the length of $15km$, containing 4 lanes, 2 lanes for each direction. Vehicles enter the road with different traffic flow rates: 600, 900, 1200, 1500, 1800 and 2100 *vehicles per hour*. The set of maximum lane speed (MLS) were specified as follows: 5, 8, 10, 12, 15 and $18m/s$ (18, 28.8, 36, 43.2 and $64.8km/s$) respectively. Vehicle number is set to 100, and 50 vehicles for each direction. According to the combination of the maximum lane speed and the traffic flow rate, SUMO generated 36 traffic traces. The main simulation parameters are given in Table III.

TABLE III
SIMULATION PARAMETERS

Parameter	Value
Simulation time	$800s$
Beacon Interval	$2.0s$
T_{beacon}	$5s$
Road length	$15km$
Length of car	$5m$
Maximum lane speed	5-$18m/s$
Traffic flow rate	600-2100 $vehicles/hour$
Number of vehicles	50 for each direction
D_t	100-$200m$
Transmission Range	$200m$
MAC protocol	IEEE 802.11p
Bandwidth	$10MHz$
Propagation model	Two-ray Ground

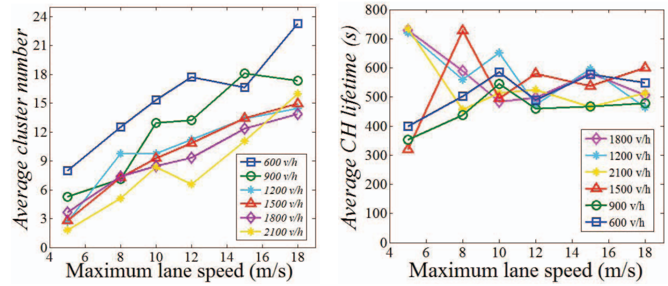
B. Results and analysis

We evaluate the cluster performance of our proposed mechanism using the following two metrics.

1) Average number of clusters: average number of clusters on the road.
2) Average CH lifetime: average duration time that a vehicle remains the cluster head during the simulation.

The *average number of clusters* allows us evaluating the quality of cluster formation. In the worst scenario, each vehicle represents a single cluster and shows no interest. The *average CH lifetime* represents the cluster's lifetime. In general, a longer lifetime of CH represents a more stable cluster. In the first simulation phase, we evaluate the influence of maximum lane speed (MLS) and traffic flow rate on clustering performance. In the second simulation phase, we analyze the D_t effects on clustering performance. The simulation results are analyzed in the following section.

1) Maximum lane speed effects: We run our clustering algorithm under the scenarios with different combinations of MLS and traffic flow rate. The transmission range of each vehicle is set to $200m$.The parameter D_t is set to $180m$. From Fig. 5(a), we can see that the average number of clusters increases quickly with the increased maximum lane speed. In our proposed algorithm, vehicles join and leave a cluster more frequently with the increased vehicle velocity. A leaving vehicle could claim itself as a new CHt and begins a new cluster formation procedure if it does not find another cluster to join. Therefore, the number of clusters on the road increases rapidly. In this case, some clusters are not robust enough to the changes of maximum lane speed. In addition, cluster merging procedure happens sometimes during the simulation, causing the cluster number decreases. From the results, we notice that when traffic flow rate is 2100 $vehicles/h$, average cluster number decreases when MLS is 12 m/s.

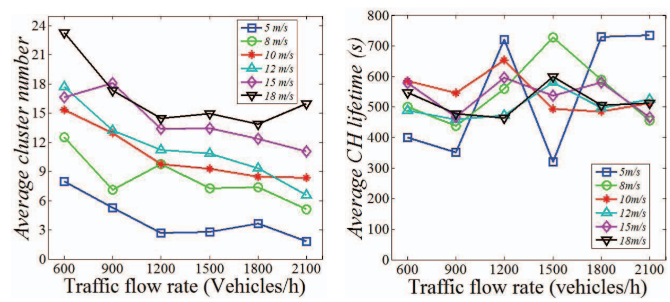


(a) Average cluster number (b) Average CH lifetime

Fig. 5. Maximum lane speed effects

Fig. 5(b) shows the influence of MLS on the average CH lifetime. We can see that with the increased MLS, the average CH lifetime is becoming more and more stable, between $450s$ and $600s$. When the maximum lane speed is $5m/s$, fewer clusters are presented on the road according to Fig. 5(a). Therefore, the average CH lifetime deviation for different traffic flow rates is high. When the average cluster number increases, this deviation becomes smaller. Moreover, we can estimate that the average value of the results in Fig. 5(b) is relatively stable, from 500-$550s$. It is an interesting result, because the CH only changes its state when a cluster merging happens or all of its CMs leave.

2) Traffic flow rate effects: Fig. 6(a) presents the traffic flow rate effects on the average number of clusters. The average number of clusters decreases when the traffic flow rate increases. Our cluster size is controlled by $2D_t$, a higher traffic flow rate means the higher vehicle density and the less clusters on the road.

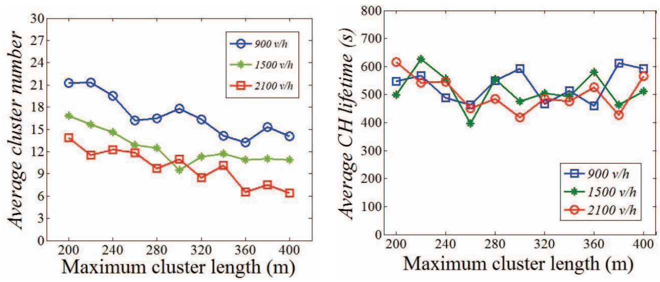


(a) Average cluster number (b) Average CH lifetime

Fig. 6. Traffic flow rate effects

Fig. 6(b) presents the traffic flow rate effects on the average CH lifetime. As we imagined, the average CH lifetime is relatively stable in the range between $350s$ to $750s$. We also estimate the average value of the results in Fig. 6(b), the average CH lifetime is also relatively stable.

3) Cluster length effects: In the second simulation, we evaluate the D_t effects on our clustering performance. The maximum lane speed is set to $12m/s$ and the set of traffic flow rate is 900, 1500 and 2100 vehicles per hour. The parameter D_t varies from $100m$ to $200m$. In Fig. 7(a), we can see that the average cluster number decreases with the growth of D_t. When the traffic flow rate is constant, a cluster can contain more vehicles if the cluster length grows. Therefore, fewer average number of clusters are presented on the road.



(a) Average cluster number (b) Average CH lifetime

Fig. 7. Maximum cluster length effects

According to the results in Fig. 7(b), the average CH lifetime is always in the range between $400s$ to $600s$. The same as what we have estimated above, the estimation shows a stable average CH lifetime with the increased maximum cluster length limit, as what we have expected.

V. Conclusion

According to dynamic mobility of VANETs, vehicles can be combined together in order to construct a more stable backbone for information dissemination. Clustering algorithm, which constructs a hierarchical network architecture, has shown an efficient way for information dissemination. In this paper, we have proposed a new mobility-based clustering algorithm. The essence of our approach is to use mobility metrics for cluster formation, and to choose the central node as a cluster head. To further improve our cluster scheme, our future work will focus on the following aspects:

- Proposing a mechanism for direction prediction at intersection;
- Using real city map to evaluate our clustering mechanism;
- Comparing the cluster performance with other mobility-based clustering algorithms, like MOBIC, PPC and ALM.

References

[1] Y. Z. Hassnaa Moustafa, *Vehicular Networks: Techniques, Standards, and Applications*, 2nd ed. Auerbach Publications, 2009.

[2] "Etsi tr 102 638 v1.1.1, intelligent transport systems (its); vehicular communications; basic set of applications; definitions."

[3] M. Togou, A. Hafid, and L. Khoukhi, "Scrp: Stable cds-based routing protocol for urban vehicular ad hoc networks," *Intelligent Transportation Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–10, 2016.

[4] J. Yu and P. Chong, "A survey of clustering schemes for mobile ad hoc networks," *Communications Surveys Tutorials, IEEE*, vol. 7, no. 1, pp. 32–48, First 2005.

[5] M. Chatterjee, S. K. Das, and D. Turgut, "An on-demand weighted clustering algorithm (wca) for ad hoc networks," in *Global Telecommunications Conference, 2000. GLOBECOM'00. IEEE*, vol. 3. IEEE, 2000, pp. 1697–1701.

[6] F. D. Tolba, D. Magoni, and P. Lorenz, "Connectivity, energy and mobility driven clustering algorithm for mobile ad hoc networks," in *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE*. IEEE, 2007, pp. 2786–2790.

[7] Z. Wang, L. Liu, M. Zhou, and N. Ansari, "A position-based clustering technique for ad hoc intervehicle communication," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 38, no. 2, pp. 201–208, 2008.

[8] Ö. Kayiş and T. Acarman, "Clustering formation for inter-vehicle communication," in *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*. IEEE, 2007, pp. 636–641.

[9] R. S. Bali, N. Kumar, and J. J. Rodrigues, "Clustering in vehicular ad hoc networks: Taxonomy, challenges and solutions," *Vehicular Communications*, vol. 1, no. 3, pp. 134 – 152, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2214209614000217

[10] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of sumo–simulation of urban mobility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, 2012.

[11] V. Vèque, F. Kaisser, C. Johnen, and A. Busson, "CONVOY: a new Cluster-based routing Protocol for Vehicular Networks," in *Models and algorithms for vehicular networks*, H. Labiod and A. Beylot, Eds. ISTE Publishing Knowledge /John Wiley and Sons Inc, May 2013, pp. 91–102, chapter 4. [Online]. Available: https://hal.inria.fr/hal-00821742

[12] N. Maslekar, J. Mouzna, H. Labiod, M. Devisetty, and M. Pai, "Modified c-drive: Clustering based on direction in vehicular environment," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, June 2011, pp. 845–850.

[13] P. Basu, N. Khan, and T. D. Little, "A mobility based metric for clustering in mobile ad hoc networks," in *Distributed computing systems workshop, 2001 international conference on*. IEEE, 2001, pp. 413–418.

[14] E. Souza, I. Nikolaidis, and P. Gburzynski, "A new aggregate local mobility clustering algorithm for vanets," in *Communications (ICC), 2010 IEEE International Conference on*, May 2010, pp. 1–5.

[15] Z. Zhang, A. Boukerche, and R. Pazzi, "A novel multi-hop clustering scheme for vehicular ad-hoc networks," in *Proceedings of the 9th ACM International Symposium on Mobility Management and Wireless Access*, ser. MobiWac '11. New York, NY, USA: ACM, 2011, pp. 19–26. [Online]. Available: http://doi.acm.org/10.1145/2069131.2069135

[16] B. Hassanabadi, C. Shea, L. Zhang, and S. Valaee, "Clustering in vehicular ad hoc networks using affinity propagation," *Ad Hoc Networks*, vol. 13, Part B, pp. 535 – 548, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870513002278

[17] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007. [Online]. Available: http://www.sciencemag.org/content/315/5814/972.abstract

[18] M. Almalag and M. Weigle, "Using traffic flow for cluster formation in vehicular ad-hoc networks," in *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, Oct 2010, pp. 631–636.

[19] A. Daeinabi, A. G. P. Rahbar, and A. Khademzadeh, "Vwca: An efficient clustering algorithm in vehicular ad hoc networks," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 207 – 222, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804510001384

[20] M. Morales, C. S. Hong, and Y. C. Bang, "An adaptable mobility-aware clustering algorithm in vehicular networks," in *Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific*, Sept 2011, pp. 1–6.